IN THE CLAIMS:

Applicants amended claims 1, 6, 11, 16, 21, 25, and 30. Applicants neither added any claims nor canceled any claims.

The listing of claims replaces all prior versions, and listings, of claims in the application.

Listing of the Claims:

1. (Currently Amended) A computer implemented method comprising:

calling ~~with~~ a single macro instruction operation from a first processor, the single macro instruction operation representing a plurality of primitive security operations, the single macro instruction operation selected from a group of macro operations including a first key exchange macro operation, a second key exchange macro operation, a finish macro operation, and a server full handshake macro operation, wherein the first key exchange macro operation represents a primitive security operation including a decrypt operation, a group of modular arithmetic operations, and seventy-eight hash operations, the second key exchange operation macro represents a primitive security operation including one decrypt operation, a group of modular arithmetic operations, and twenty-two hash operations, the finish operation macro represents a primitive security operation including one decrypt operation, an encrypt operation, twelve hash operations, and the server full handshake operation macro represents a primitive security

operation including a decrypt operation, two encrypt operations, a set of modular arithmetic operations, and thirty-five hash operations;

executing the plurality of primitive security operations at a second processor in response to receiving the single macro instruction operation from the first processor, the second processor having a plurality of execution units that each can perform the single macro instruction operation, wherein a single execution unit of said plurality of execution units performs the plurality of primitive security operations that correspond to the single macro instruction;

generating a set of data from executing the plurality of primitive security operations at the second processor; and

establishing a secure session with the set of data.

2. (Original) The computer implemented method of claim 1 wherein the set of data comprises:

a set of decrypted data;

a set of encrypted data; and

a set of hashed messages.

3. (Original) The computer implemented method of claim 2 further comprising a set of random numbers.

4. (Original) The computer implemented method of claim 1 further comprising the first processor calling a second operation to establish a second secure session.

5. (Original) The computer implemented method of claim 1 wherein the secure session is an SSL 3.0 session, a TLS session, or an IPSec session.

6. (Currently Amended) A computer implemented method comprising:

calling a single macro security operation from a first processor to a second processor, the single macro security operation representing a set of primitive security operations, wherein the single macro security operation is a server full handshake macro operation;

performing the set of primitive security operations in response to the single macro security operation, the set of primitive security operations comprising,

generating a secret and a key material,

creating a first finished hash for a client message,

creating a second finished hash for a server message,

creating a finished message; and

establishing a secure session.

7. (Original) The computer implemented method of claim 6 wherein the set of operations further comprises

decrypting a pre-master secret; and

decrypting a client finished message.

8. (Original) The computer implemented method of claim 6 wherein the set of operations further comprises generating a set of random numbers.

9. (Original) The computer implemented method of claim 6 wherein the set of operations further comprises creating an expected finished message.

10. (Original) The computer implemented method of claim 6 further comprising calling a second macro security operation to establish a second secure session.

11. (Previously Presented) A system comprising:

a first network element to request a secure session; and

a second network element networked to the first network element, the second network element to call a macro security operation from a first processor, the macro security operation associated with a plurality of primitive security operations, to execute the plurality of primitive security operations at a second processor in response to the macro security operation, and to generate a set of data from the execution of the plurality of primitive security operations in response to the macro security operation selected from a group of macro operations including a first key exchange macro operation, a second key exchange macro operation, a finish macro operation, and a server full handshake macro operation, wherein the first key exchange macro operation associated with the plurality of primitive security operations including a decrypt operation, a group of modular arithmetic operations, and seventy-eight hash operations, the second key exchange macro operation associated with the plurality of primitive security operations including one decrypt operation, a group of modular arithmetic operations, and twenty-two hash operations, the finish macro operation associated with the plurality of primitive security operations including one decrypt operation, an encrypt operation, twelve hash operations, and the server full handshake macro operation associated with the plurality of primitive security operations including a decrypt operation, two encrypt operations, a set of modular arithmetic operations, and thirty-five hash operations.

12. (Original) The system of claim 11 wherein the set of data comprises:

a set of decrypted data;

a set of encrypted data; and

a set of hashed data.

13. (Original) The system of claim 11 wherein the first network element to request the secure session comprises the first network element to transmit a set of messages to the second network element, to execute a second macro security operation, and to generate a second set of data from the execution of the second macro security operation.

14. (Original) The system of claim 11 further comprising a third network element networked to the second network element, the third network element to request a second secure session with the second network element.

15. (Original) The system of claim 11 further comprising:

the first network element to request a second secure session with the second network element; and

the second network element to execute a second macro security operation to establish the second secure session with the first network element.

16. (Previously Presented) An apparatus comprising:

a first processor to call a macro security operation associated with a plurality of primitive security operations to establish a secure session, the macro security operation selected from a group of macro security operations including a first key exchange macro security operation, a second key exchange macro security operation, a finish macro security operation, and a server full handshake macro security operation,

wherein the first key exchange operation associated with the plurality of primitive security operations including a decrypt operation, a group of modular arithmetic operations, and seventy-eight hash operations, the second key exchange operation associated with the plurality of primitive security operations including one decrypt operation, a group of modular arithmetic operations, and twenty-two hash operations, the finish operation macro represents one decrypt operation, an encrypt operation, twelve hash operations, and the server full handshake operation macro associated with the plurality of primitive security operations including a decrypt operation, two encrypt operations, a set of modular arithmetic operations, and thirty-five hash operations;

a second processor coupled to the first processor, the second processor to perform the plurality of primitive security operations in response to the macro security operation from said first processor; and

a memory coupled to the first and the second processor, the memory to store a set of data generated by the second processor.

17. (Original) The apparatus of claim 16 wherein the second processor comprises:

a request unit to fetch and to distribute the macro security operation; and

a plurality of execution units coupled to the request unit, one of the plurality of execution units to execute the plurality of primitive security operations.

18. (Original) The apparatus of claim 17 further comprising:

the first processor to call a second macro security operation after calling the first macro security operation; and

a second one of the plurality of execution units to execute a second plurality of primitive security operations corresponding to the second macro security operation before the one of the plurality of execution units completes execution of the plurality of primitive security operations.

19. (Original) The apparatus of claim 17 wherein the one of the plurality of execution units comprises:

a microcode unit to translate the macro security operation into a plurality of primitive security operations;

an execution queue unit coupled to the microcode unit, the execution queue unit to queue the plurality of primitive security operations;

a plurality of primitive security operation units coupled to the execution queue unit, the plurality of primitive security operation units to perform the plurality of primitive security operations; and

a bus coupled to the plurality of primitive security operation units, the bus to transmit data.

20. (Original) The apparatus of claim 16 further comprising the memory to store a set of source data.

21. (Currently Amended) An apparatus comprising:

a first processor to give the command for a macro security operation associated with a plurality of primitive security operations, the macro security operation selected from a group including a key exchange operation macro, a finish operation macro, and a server full handshake operation macro, wherein the key exchange operation macro associated with the plurality of primitive security operations including

one decrypt operation, a group of modular arithmetic operations, and twenty-two hash operations, the key exchange operation macro associated with the plurality of primitive security operations including a decrypt operation, a group of modular arithmetic operations, and seventy-eight hash operations, the finish operation macro associated with the plurality of primitive security operations including one decrypt operation, an encrypt operation, twelve hash operations, and the server full handshake operation macro associated with the plurality of primitive security operations including a decrypt operation, two encrypt operations, a set of modular arithmetic operations, and thirty-five hash operations;

a second processor coupled to the first processor, the second processor comprising

a request unit to retrieve the macro security operation associated with the plurality of primitive security operations,

a plurality of execution units coupled to the request unit, one of the plurality of execution units to perform the plurality of primitive security operations retrieved by the request unit, the plurality of primitive security operations corresponding to the macro security operation; and

a memory coupled to the first and second processor, the memory to store a set of data generated by the second processor.

22. (Original) The apparatus of claim 21 further comprising the memory to store a set of source data from the host processor.

23. (Original) The apparatus of claim 21 wherein each of the plurality of execution units comprises:

a microcode unit to translate the macro security operation into the plurality of primitive security operations;

an execution queue unit coupled to the microcode unit, the execution queue unit to queue the plurality of primitive security operations;

a plurality of primitive security operation units coupled to the execution queue unit, the plurality of primitive security operation units to perform the plurality of primitive security operations; and

a bus coupled to the plurality of primitive security operation units, the bus to transmit the set of generated data.

24. (Original) The apparatus of claim 21 further comprising:

the first processor to call a primitive security operation; and

a second one of the plurality of execution units to execute the primitive security operations.

25. (Currently Amended) A machine-readable medium that provides instructions, which when executed by a set of one or more processors, cause said set of processors to perform operations comprising:

executing a macro security operation at a first one of the set of processors, the macro security operation associated with a plurality of primitive security operations, the macro security operation selected from a group including a key exchange macro, a finish macro, and a server full handshake macro, wherein the key exchange macro associated with the plurality of primitive security operations including one decrypt operation, a group of modular arithmetic operations, and twenty-two

hash operations, the key exchange operation macro associated with the plurality of primitive security operations including a decrypt operation, a group of modular arithmetic operations, and seventy-eight hash operations, the finish macro associated with the plurality of primitive security operations including one decrypt operation, an encrypt operation, twelve hash operations, and the server full handshake macro associated with the plurality of primitive security operations including a decrypt operation, two encrypt operations, a set of modular arithmetic operations;

executing the plurality of primitive security operations at a second one of the set of processors in response to the macro security operation;

generating a set of data from executing the plurality of primitive security operations in response to the macro security operation; and

establishing a secure session with the set of data.

26. (Original) The machine-readable medium of claim 25 wherein the set of data comprises:

a set of decrypted data;

a set of encrypted data; and

a set of hashed messages.

27. (Original) The machine-readable medium of claim 26 wherein the set of data further comprises a set of random numbers.

28. (Original) The machine-readable medium of claim 25 further comprising the first processor calling a second operation to establish a second secure session.

29. (Original) The machine-readable medium of claim 25 wherein the secure session is an SSL 3.0 session, a TLS session, or an IPSec session.

30. (Previously Presented) A machine-readable medium that provides instructions, which when executed by a set of one or more processors, cause said set of processors to perform operations comprising:

calling a single macro security operation from a first one of the set of processors, the single macro security operation associated with a set of primitive security operations, wherein the single macro instruction is a server full handshake macro instruction;

performing the set of primitive security operations at a second one of the set of processors in response to the single macro security operation, the set of primitive security operations comprising,

generating a secret and a key material,

creating a first finished hash for a client message,

creating a second finished hash for a server message,

creating a finished message; and

establishing a secure session.

31. (Original) The machine-readable medium of claim 30 wherein the set of operations further comprises decrypting a pre-master secret and a client finished message.

32. (Original) The machine-readable medium of claim 30 wherein the set of operations further comprises generating a set of random numbers.

33. (Original) The machine-readable medium of claim 30 the set of operations further comprising creating an expected finished message.

34. (Original) The machine-readable medium of claim 30 further comprising calling a second macro security operation to establish a second secure session.